

---

# **Programming and Interfacing the 8051 Microcontroller in C and Assembly**

---

Sencer Yeralan, P.E., Ph.D.  
Helen Emery

Rigel Press, a Division of Rigel Corporation

# Table of Contents

<b>TABLE OF CONTENTS .....</b>	<b>III</b>
<b>ACKNOWLEDGEMENTS.....</b>	<b>IX</b>
<b>PREFACE .....</b>	<b>XI</b>
<b>CHAPTER 1 ARCHITECTURE.....</b>	<b>1</b>
1.1. MICROPROCESSORS AND MICROCONTROLLERS .....	1
1.2. THE 8051 MICROCONTROLLER FAMILY ARCHITECTURE .....	2
1.2.1. On-Chip Memory.....	6
1.2.2. Using the On-Chip Facilities .....	12
<b>CHAPTER 2 ASSEMBLY LANGUAGE PROGRAMMING .....</b>	<b>27</b>
2.1. BACKGROUND .....	27
2.2. DATA TRANSFER INSTRUCTIONS.....	33
2.2.1. The Immediate Addressing Mode .....	35
2.2.2. The Direct Addressing Mode .....	37
2.2.3. The Register Addressing Mode .....	39
2.2.4. The Register-Specific Addressing Mode .....	40
2.2.5. The Register Indirect Addressing Mode .....	41
2.2.6. The Register Indexed Addressing Mode .....	44
2.2.7. Stack-Oriented Data Transfer .....	45
2.2.8. Exchange Instructions .....	46
2.2.9. Bit-Oriented Data Transfer.....	47
2.3. DATA PROCESSING INSTRUCTIONS .....	50
2.3.1. Arithmetic Instructions .....	50
2.3.2. Logic Instructions .....	60
2.4. PROGRAM FLOW CONTROL INSTRUCTIONS.....	66
2.4.1. Unconditional Jump Instructions .....	67
2.4.2. Conditional Jump Instructions.....	71
2.4.3. Call and Return Instructions .....	75
2.4.4. Subroutines and Interrupt Service Routines.....	78
2.5. ENHANCED MEMBERS OF THE 8051 FAMILY .....	83
2.5.1. Fast Microcontrollers .....	84
2.5.2. Enhanced Memory Options .....	84
2.5.3. Additional Data Pointers.....	85
2.5.4. The Multiply and Divide Unit of the 80C517 .....	87
2.5.5. 10-Bit Analog-to-Digital Conversion .....	89
2.5.6. The I <sup>2</sup> C Bus Support .....	89

2.5.7. ARCNET Token Bus Network Support.....	90
2.5.8. Controller Area Network (CAN) Bus Support.....	91
2.5.9. Universal Serial Bus (USB) Support.....	92
2.5.10. System-On-Chip 8051's.....	92
2.5.11. Migrating to 16-Bit Architectures .....	93
2.5.12. Information Sources .....	93
<b>CHAPTER 3 ASSEMBLY LANGUAGE TECHNIQUES.....</b>	<b>95</b>
3.1. INTRODUCTION .....	95
3.2. DATA BLOCK TRANSFER ROUTINES .....	95
3.3. TABLE LOOK UP PROCEDURES.....	96
3.4. ASCII CONVERSION ROUTINES.....	99
3.5. JUMP TABLES .....	103
3.6. N-WAY BRANCHING .....	107
3.7. SIGNED ARITHMETIC ROUTINES.....	109
3.7.1. Two's Complement Conversion Routines.....	110
3.7.2. 8-Bit Signed Addition and Subtraction.....	113
3.7.3. 8-Bit Signed Multiplication and Division.....	114
3.8. 16-BIT UNSIGNED ARITHMETIC ROUTINES .....	115
3.8.1. 16-Bit Addition and Subtraction.....	115
3.8.2. 16-Bit Unsigned Multiplication and Division.....	116
3.9. 16-BIT SIGNED ARITHMETIC ROUTINES .....	123
3.9.1. 16-Bit Signed Addition and Subtraction.....	123
3.9.2. 16-Bit Signed Multiplication and Division.....	124
3.10. FLOATING-POINT ARITHMETIC ROUTINES.....	125
3.10.1. Floating-Point Multiplication and Division .....	125
3.11. PSEUDO-RANDOM NUMBERS.....	132
3.12. STRING MANIPULATION ROUTINES .....	133
3.13. SOFTWARE TIMING ROUTINES .....	139
3.14. RE-ENTRANT ROUTINES.....	141
3.15. SOFTWARE STACKS .....	143
3.16. MACRO DEFINITIONS, PREPROCESSOR DIRECTIVES, AND CONDITIONAL ASSEMBLY .....	144
<b>CHAPTER 4 INTRODUCTORY ASSEMBLY LANGUAGE EXPERIMENTS.....</b>	<b>149</b>
4.1 INPUT/OUTPUT OPERATIONS.....	150
4.2. A SUBROUTINE FOR DRIVING SEVEN-SEGMENT DISPLAYS .....	151
4.3. SERIAL COMMUNICATIONS .....	154
4.4. INTERRUPT SERVICE ROUTINES.....	158
4.4.1. Using an Interrupt Vector Table to Redirect the Interrupts .....	161
4.4.2. Using Multiple Interrupts.....	163
4.5. COUNTER OPERATIONS .....	164
4.6. TIMER OPERATIONS .....	168

4.7. USING THE PTR A UNIT COMPARE AND CAPTURE FUNCTIONS .....	171
4.7.1. Pulse Train Generation with the PTR A Unit .....	171
4.7.2. Pulse Width Modulation.....	174
4.7.3. Pulse Width and Period Measurement.....	176
4.7.4. Software Timers Using the PTR A Unit.....	184
4.8. ANALOG-TO-DIGITAL CONVERTER .....	185
4.8.1. Interrupt-Driven Timer and Analog-to-Digital Conversion Operation .....	187
4.8.2. 10-Bit Resolution with the Analog-to-Digital Conversion.....	194
4.9. SERIAL COMMUNICATIONS .....	201
<b>CHAPTER 5 C LANGUAGE PROGRAMMING.....</b>	<b>207</b>
5.1. C FOR MICROCONTROLLERS.....	207
5.1.1. In-Line Assembly.....	207
5.1.2. C Extensions.....	208
5.1.3. The Reads51 SmallC-Compatible Compiler.....	210
5.1.3.1. Mixed C / Assembly Module Projects.....	211
5.2. INTRODUCTORY C LANGUAGE EXPERIMENTS.....	212
5.2.1. Input/Output Operations .....	213
5.2.2. A C Function for Driving Seven-Segment Displays .....	214
5.2.3. Serial Communications.....	215
5.2.4. Interrupt Service Routines .....	223
5.2.5. Timer and Counter Operations .....	224
5.2.6. Analog-to-Digital Conversion .....	227
5.3. MULTITASKING AND MULTITHREADING .....	229
5.3.1. A Real-Time Multitasking Utility: RTMT.....	231
5.3.2. Using the RTMT Utility.....	239
5.3.3. Inter-Task Synchronization .....	245
<b>CHAPTER 6 HARDWARE EXPERIMENTS.....</b>	<b>249</b>
6.1. IMPLEMENTING SIMPLE BOOLEAN FUNCTIONS.....	249
6.1.1. Combinational Logic.....	250
6.1.2. Sequential Logic.....	252
6.2. SCANNING A KEYPAD .....	254
6.3. STEPPER MOTOR CONTROL.....	268
6.4. A FREQUENCY GENERATOR WITH TIMERS.....	278
6.5. MEASURING TEMPERATURE AND LIGHT INTENSITY WITH AN ANALOG-TO-DIGITAL CONVERTER .....	282
6.6. DIGITAL-TO-ANALOG CONVERSION .....	291
6.6.1. Pulse Width Modulation.....	292
6.6.2. Using Digital-to-Analog Converter Chips.....	300
6.7. DC MOTOR SPEED REGULATION WITH A FEEDBACK SYSTEM .....	304
6.8. INTELLIGENT PERIPHERALS - INTERFACING WITH LIQUID CRYSTAL DISPLAYS.....	313
6.9. IMPLEMENTING A MULTI-DROP RS-485 NETWORK .....	336

6.9.1. A Simple Network Protocol .....	341
<b>CHAPTER 7 BUILDING AN 8051-BASED MICROCONTROLLER SYSTEM.....</b>	<b>349</b>
7.1. HARDWARE .....	349
7.1.1. An 8051-Based System .....	349
7.1.2. 8051-Based System Bill of Materials .....	354
7.1.3. An 80C515-Based System.....	357
7.1.4. 80C515-Based System Bill of Materials .....	361
7.2. INTERFACING THE MICROCONTROLLER SYSTEM .....	363
7.3. FIRMWARE .....	368
7.3.1. MINMON - A Minimal 8051 Monitor Program .....	368
7.3.2. Setting Up an Interrupt Vector Table .....	383
7.4. USING THE MICROCONTROLLER SYSTEM TO RUN THE EXAMPLE PROGRAMS .....	385
<b>CHAPTER 8 DEVELOPING MICROCONTROLLER APPLICATIONS .....</b>	<b>387</b>
8.1. SUGGESTED APPLICATIONS .....	387
8.2. THE SOFTWARE DEVELOPMENT CYCLE .....	388
<b>BIBLIOGRAPHY .....</b>	<b>391</b>
.....	<b>391</b>
<b>APPENDIX A READS51 SOFTWARE .....</b>	<b>395</b>
<b>APPENDIX B A BRIEF REVIEW OF THE C PROGRAMMING LANGUAGE .....</b>	<b>397</b>
B.1 C LANGUAGE PHILOSOPHY .....	397
B.1.1 Ingredients of a C Program .....	398
B.1.2 Code Appearance and Style.....	398
B.2 FUNCTIONS.....	399
B.3 VARIABLES .....	399
B.3.1 Scalars .....	399
B.3.2 Pointers .....	400
B.3.3 Arrays .....	401
B.4 CONSTANTS.....	402
B.4.1 Compiler Directives .....	402
B.5 STATEMENTS .....	402
B.5.1 Compound Statements.....	402
B.5.2 Expressions .....	403
B.5.3 Conditions.....	404
B.5.4 Iteration Statements .....	404
B.5.5 Selection Statements .....	408
B.6 COMMENTS.....	409
B.7 STANDARD (RUN TIME) LIBRARIES .....	410

B.8 REFERENCES.....	410
<b>APPENDIX C NUMBER SYSTEMS.....</b>	<b>411</b>
C.1 BINARY NUMBERS .....	411
C.2 HEXADECIMAL NUMBERS .....	412
C.3 BINARY-CODED DECIMAL NUMBERS.....	413
C.4 TWO'S-COMPLEMENT CONVENTION FOR SIGNED NUMBERS.....	413
<b>APPENDIX D 8051 TOOLS AND INFORMATION SOURCES.....</b>	<b>417</b>
D.1. DIRECTORIES AND PERIODICALS .....	417
D.2. CHIP MANUFACTURERS .....	418
D.3. SOFTWARE VENDORS.....	420
D.4. HARDWARE VENDORS .....	425
D.4.1. Single-Board Computers.....	425
D.4.2. Parts Suppliers .....	427
<b>INDEX .....</b>	<b>429</b>